IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of:	Confirmation No.: 7027
Jack B. DENNIS	Art Unit: 2664
Application No.: 09/715,778	Examiner: Kevin D. Mew
Filed: November 17, 2000	Atty. Docket: 2222.4220001
For: **Prioritizing Resource Utilization In Multi-Thread Computing System**	

Brief on Appeal Under 37 C.F.R. § 41.37

Mail Stop Appeal Brief - Patents

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

A Notice of Appeal from the final rejection of claims 1 - 30 was filed on October 20, 2005. Appellant hereby files one copy of this Appeal Brief, together with the required fee set forth in 37 C.F.R. § 41.20(b)(2).

It is not believed that extensions of time are required beyond those that may otherwise be provided for in documents accompanying this paper. However, if additional extensions of time are necessary to prevent abandonment of this application, then such extensions of time are hereby petitioned under 37 C.F.R. § 1.136(a), and any fees required therefor (including fees for net addition of claims) are hereby authorized to be charged to our Deposit Account No. 19-0036.

05/23/2006 JADD01 00000120 09715778
02 FC:1402 500.00 OP

Table of Contents

I. Real Party In Interest (37 C.F.R. § 41.37(c)(1)(i))

The real party in interest in this appeal is the United States Government, as represented by the Secretary of the Navy. In addition, Nanocomm Systems, LLC, located at 2215-B Renaissance Drive, Suite 5, Las Vegas, NV 89119, is a licensee under this application.

II. Related Appeals and Interferences (37 C.F.R. § 41.37(c)(1)(ii))

To the best of the knowledge of Appellant, Appellant's legal representative, and Appellant's assignee, there are no other appeals or interferences which will directly affect or be directly affected by or have a bearing on a decision by the Board of Patent Appeals and Interferences ("the Board") in the pending appeal.

III. Status of Claims (37 C.F.R. § 41.37(c)(1)(iii))

This application was originally filed as U.S. Non-Provisional Application No. 09/715,778 on November 17, 2000 with 30 claims. This application is a continuation of U.S. Provisional Application No. 60/166,685 filed on November 19, 1999. In response to an Office Action mailed September 20, 2004, Appellant filed an Amendment and Reply on January 4, 2005 in which claims 1, 2, 5-12, 15-22, and 25-30 were amended. The Examiner issued a Final Office Action on June 10, 2005, to which Appellant filed a Reply on August 10, 2005, in which no amendments were made. The Examiner subsequently mailed an Advisory Action on September 20, 2005.

Claims 1-30 are pending. Claims 1-30 are rejected and are being appealed. A copy of the claims on appeal can be found in the attached Appendix as required under 37 C.F.R. § 41.37(c)(1)(viii).

IV. Status of Amendments (37 C.F.R. § 41.37(c)(1)(iv))

No amendments have been filed subsequent to the Final Office Action dated June 10, 2005. All amendments presented in the Amendment and Reply dated January 4, 2005 have been entered.

V. Summary of Claimed Subject Matter (37 C.F.R. § 41.37(c)(1)(v))

A concise explanation of the invention is provided below for each of the independent claims involved in the appeal. The explanation refers to the specification by page and line number, and to the drawings, if any, by reference characters.

For each independent claim involved in the appeal and for each dependent claim argued separately under the provisions of paragraph (c)(1)(vii), every means plus function and step plus function as permitted by 35 U.S.C. § 112, sixth paragraph, are identified. The structure, material, or acts described in the specification as corresponding to each claimed function are set forth with reference to the specification by page and line number, and to the drawings, if any, by reference characters.

FIG. 1 of the Specification illustrates a multi-processor core 110 interfacing with peripheral units 130 in accordance with the subject matter claimed. The multi-processor core 110 contains multiple processors 210 as shown in FIG. 2. Independent claim 1 is directed to an apparatus 100 comprising a processor 210 capable of simultaneous execution of two or more threads of instructions. (Specification, p. 8, ll. 7-9). The processor of claim 1 includes at least one resource unit capable of being assigned to two or more of the threads. (*see, e.g.,* Functional Unit 1 450, Functional Unit 2 450, Memory Access Unit 460, Peripheral Unit Interface 420, FIG. 6). The processor further includes a priority register 580 which stores priority code information for each thread of instructions. (Specification, p. 12, ll. 20-25). A priority selector 610 uses the priority code information from the priority register 580 in order to allocate the resource unit (Functional Unit 1 450, Functional Unit 2 450, Memory Access Unit 460, Peripheral Unit Interface 420, etc.) to a thread of instructions. (Specification, p. 13, ll. 21-29).

Independent claims 11 and 21, as well as their dependent claims, find similar support to the above within the specification.

VI. Grounds of Rejection to be Reviewed on Appeal (37 C.F.R. § 41.37(c)(1)(vi))

The Examiner finally rejected claims 1-30 under 35 U.S.C. § 102(e) as allegedly being anticipated by U.S. Patent No. 6,434,590 to Blelloch et al. ("Blelloch").

Accordingly, the sole ground of rejection to be reviewed on appeal is:

Whether claims 1-30 would have been anticipated by U.S. Patent No. 6,434,590 to Blelloch under 35 U.S.C. § 102(e).

VII. Argument (37 C.F.R. § 41.37(c)(1)(vii))

The sole ground of rejection to be reviewed on appeal is whether claims 1-30 would have been anticipated by U.S. Patent No. 6,434,590 to Blelloch under 35 U.S.C. § 102(e).

A. Rejection of claims 1-30 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,434,590 to Blelloch

A Final Office Action was mailed on June 10, 2005, rejecting claims 1-30 under 35 U.S.C. § 102(e) as allegedly being anticipated by Blelloch. Appellant's remarks focus mainly on independent claims 1, 11, and 21, because any claim which depends from a patentable independent claim is also patentable at least by virtue of its dependency.

In proceedings before the Patent and Trademark Office, the examiner bears the burden of establishing a *prima facie* case of anticipation based on prior art which must disclose each limitation of the claims. *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631 (Fed. Cir. 1987). Because the Blelloch reference fails to disclose each limitation of the claims, the rejection of claims 1-30 must be reversed.

1. The Anticipation Rejection with Respect to Claims 1-10 is in Error and Must be Reversed

Blelloch does not teach or suggest each feature of Appellant's independent claim 1. Independent claim 1 recites "an apparatus", wherein the apparatus includes:

> a processor capable of simultaneous execution of two or more threads of instructions, where said processor comprises:
>
> at least one resource unit capable of being assigned to two or more of the threads;
>
> a priority register to store thread information for the threads, the thread information including a priority code corresponding to each thread, at least one of the threads requesting use of the resource unit; and

> a priority selector coupled to the priority register to generate an assignment signal to assign the at least one resource unit to the requesting thread according to the priority codes.

Blelloch is directed to an assignment manager operable to provide tasks to a group of processing elements based on priority information. (Blelloch, col. 1, ll. 51-59). The processing elements contain a task buffer with tasks to be performed, and fill the task buffer with tasks from the assignment manager. (Blelloch, col. 2, ll. 52-55). The processing elements continue to process the tasks, and create new tasks, until a specified criterion has been met. (Blelloch, col. 3, ll. 2-5). Tasks may contain instructions that have serial dependencies on other tasks and must therefore wait for the other tasks to be processed first. Blelloch discloses determining groupings of tasks in a sequential scheduling order that are available for parallel processing. (Blelloch, col. 4, ll. 15-20). Tasks are then transmitted to each of the processing elements from the assignment manager via a router. (Blelloch, col. 4, ll. 60-64). Furthermore, after processing, the processing elements inform the assignment manager of completed tasks and new live tasks via the router. (Blelloch, col. 5, ll. 2-6).

Appellant's claim 1 and Blelloch are very different. Appellant's claim 1 recites "at least one resource unit capable of being assigned to two or more of the threads." In Blelloch, there is no indication of a resource unit capable of being assigned to any of the threads. The Examiner indicates in the Final Office Action that the router in Blelloch is the analogous resource unit. (Office Action, p. 2). However, Blelloch never makes any reference to the router being assigned to a thread, only indicating that the function of the router is to route communications between the processing elements and the assignment manager. For example, Blelloch recites:

> Each processing element PE1 then takes (draws, or extracts) a task as soon as it becomes available through the router RT1 which routes the set of tasks made available by the assignment manager AM1. (Blelloch, col. 2, ll. 48-51).
>
> In step 620, the processing element requests from the assignment manager AM1 via the router RT1 some number of tasks from the head of the tasks queue TQ1, and receives the tasks via the router RT1. (Blelloch, col. 4, ll. 61-64).
>
> At this point, in step 627, the processing element PE1 informs, via the router RT1, the assignment manager AM1 of completed tasks and new live tasks . . . (Blelloch, col. 5, ll. 2-5).

Thus, Blelloch's router is nothing more than a conduit for directing communications between a processing element and an assignment manager.

Furthermore, the Examiner suggests that the assignment manager of Blelloch is analogous to the priority selector of the present invention, which assigns the router resource to the threads. (Office Action, p. 3). Blelloch nowhere indicates that the router is assigned to a thread by the assignment manager. This would be contrary to the operation of Blelloch, since the processing elements must use the router in order to access the assignment manager, indicating that the assignment manager cannot regulate the router's use by the processing elements.

There is also no discussion of an "assignment signal" within Blelloch in order to allocate any resource to a task in a processing element. Assuming, *arguendo*, that the router in Blelloch is the allocated resource, there is then no analogous priority selector regulating the use of the router by a processing element based on an assignment signal.

Since Blelloch does not teach or suggest each and every feature of independent claim 1 it does not anticipate that claim. Accordingly, the Examiner's rejection of claim 1 must be reversed. Furthermore, dependent claims 2-10 are also not anticipated by Blelloch for at least the same reasons as independent claim 1 from which they depend

and further in view of their own respective features. Accordingly, the Examiner's rejection of claims 2-10 must also be reversed.

2. The Anticipation Rejection with Respect to Claims 11-20 is in Error and Must be Reversed

Appellant's independent claim 11 recites a method for which the apparatus of claim 1 is a physical embodiment as discussed above. Claim 11 is distinguishable over Blelloch for at least the same reasons as claim 1. Since Blelloch does not teach or suggest each and every feature of independent claim 11, it does not anticipate that claim. Accordingly, the Examiner's rejection of claim 11 must be reversed. Furthermore, dependent claims 12-20 are also not anticipated by Blelloch for at least the same reasons as independent claim 11 from which they depend and further in view of their own respective features. Accordingly, the Examiner's rejection of claims 12-20 must also be reversed.

3. The Anticipation Rejection with Respect to Claims 21-30 is in Error and Must be Reversed

Appellant's independent claim 21 recites a processor comprising similar features as independent claim 1 discussed above. Claim 21 is distinguishable over Blelloch for at least the same reasons as claim 1. Since Blelloch does not teach or suggest each and every feature of independent claim 21, it does not anticipate that claim. Accordingly, the Examiner's rejection of claim 21 must be reversed. Furthermore, dependent claims 22-30 are also not anticipated by Blelloch for at least the same reasons as independent claim 21 from which they depend and further in view of their own respective features. Accordingly, the Examiner's rejection of claims 22-30 must also be reversed.

B. Conclusion

The subject matter of claims 1-30 is patentable over the cited prior art because the Examiner has failed to make a *prima facie* case of anticipation or obviousness. Therefore, Appelant respectfully requests that the Board reverse the Examiner's final rejection of these claims under 35 U.S.C. § 102 and remand this application for issue.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.

Edward J. Kessler
Attorney for Applicant
Registration No. 25,688

Date: May 22, 2006

1100 New York Avenue, N.W.
Washington, D.C. 20005-3934
(202) 371-2600

VIII. Claims Appendix

1. An apparatus comprising:

a processor capable of simultaneous execution of two or more threads of instructions,

where said processor comprises:

at least one resource unit capable of being assigned to two or more of the threads;

a priority register to store thread information for the threads, the thread information including a priority code corresponding to each thread, at least one of the threads requesting use of the resource unit; and

a priority selector coupled to the priority register to generate an assignment signal to assign the at least one resource unit to the requesting thread according to the priority codes.

2. The apparatus of claim 1 wherein the at least one resource unit is one of an instruction fetch unit, a memory locking unit, a load unit, a store unit, an input/output unit, a peripheral unit interface, and a functional unit.

3. The apparatus of claim 2 wherein the functional unit is one of an arithmetic unit, a logic unit, and an arithmetic and logic unit.

4. The apparatus of claim 1 further comprising:

an instruction multiplexer coupled to the priority selector to pass instructions stored in a plurality of instruction registers to execution units according to the assignment signal.

5. The apparatus of claim 1 further comprising:

a priority assignor coupled to the priority register to set the thread information including at least one of the priority codes corresponding to the at least one of the threads in response to a start instruction from an instruction decoder and dispatcher.

6. The apparatus of claim 5 wherein the priority assignor sets an active flag in the priority register corresponding to the at least one of the threads in response to the start instruction.

7. The apparatus of claim 6 wherein resets the active flag in the priority register corresponding to the at least one of the threads in response to a quit instruction from the instruction decoder and dispatcher.

8. The apparatus of claim 1 wherein the priority selector assigns the at least one resource unit to the at least one of the threads if the at least one of the threads is not served and the at least one resource unit is free.

9. The apparatus of claim 8 wherein the at least one of the threads has highest priority code among a set of ready threads.

10. The apparatus of claim 8 wherein the priority selector iteratively assigns resource units to threads in the set of ready threads according to the corresponding priority codes and resource availability until the set becomes empty.

11. A method comprising:

executing two or more threads of instructions simultaneously in a processor;

storing thread information for the threads, the thread information including a priority code corresponding to each thread, at least one of the P threads requesting use of at least one resource unit capable of being assigned to two or more of the threads; and

generating an assignment signal to assign the at least one resource unit to the requesting thread according to the priority codes.

12. The method of claim 11 wherein the at least one resource unit is one of an instruction fetch unit, a memory locking unit, a load unit, a store unit, an input/output unit, a peripheral unit interface, and a functional unit.

13. The method of claim 12 wherein the functional unit is one of an arithmetic unit, a logic unit, and an arithmetic and logic unit.

14. The method of claim 11 further comprising:

passing instructions stored in a plurality of instruction registers to execution units according to the assignment signal.

15. The method of claim 11 further comprising:

setting the thread information including at least one of the priority codes corresponding to the at least one of the threads in response to a start instruction from an instruction decoder and dispatcher.

16. The method of claim 15 wherein setting the thread information comprises setting an active flag in the priority register corresponding to the at least one of the threads in response to the start instruction.

17. The method of claim 16 wherein setting the thread information comprises resetting the active flag in the priority register corresponding to the at least one of the threads in response to a quit instruction from the instruction decoder and dispatcher.

18. The method of claim 11 wherein generating the assignment signal comprises generating the assignment signal to assign the at least one resource unit to the at least one of the threads if the at least one of the threads is not served and the at least one resource unit is free.

19. The method of claim 18 wherein the at least one of the threads has highest priority code among a set of ready threads.

20. The method of claim 1 wherein generating the assignment signal comprises iteratively assigning resource units to threads in the set of ready threads according to the corresponding priority codes and resource availability until the set becomes empty.

21. A processor capable of simultaneous execution of two or more threads of instructions, comprising:

at least one a resource unit to provide resource for use by the threads, capable of being assigned to two or more of the threads; and

a resource prioritizer coupled to the resource unit to prioritize resource utilization, the resource prioritizer comprising:

a priority register to store thread information for the threads, the thread information including a priority code corresponding to each thread, at least one of the threads requesting use of the resource unit, and

a priority selector coupled to the priority register to generate an assignment signal to assign the at least one resource unit to the requesting thread according to the priority codes.

22. The processor of claim 21 wherein the at least one resource unit is one of an instruction fetch unit, a memory locking unit, a load unit, a store unit, an input/output unit, a peripheral unit interface, and a functional unit.

23. The processor of claim 22 wherein the functional unit is one of an arithmetic unit, a logic unit, and an arithmetic and logic unit.

24. The processor of claim 21 the resource prioritizer further comprising:

an instruction multiplexer coupled to the priority selector to pass instructions stored in a plurality of instruction registers to execution units according to the assignment signal.

25. The processor of claim 21 wherein the resource prioritizer further comprising:

a priority assignor coupled to the priority register to set the thread information including at least one of the priority codes corresponding to the at least one of the threads in response to a start instruction from an instruction decoder and dispatcher.

26. The processor of claim 25 wherein the priority assignor sets an active flag in the priority register corresponding to the at least one of the threads in response to the start instruction.

27. The processor of claim 26 wherein resets the active flag in the priority register corresponding to the at least one of the threads in response to a quit instruction from the instruction decoder and dispatcher.

28. The processor of claim 21 wherein the priority selector assigns the at least one resource unit to the at least one of the threads if the at least one of the P threads is not served and the at least one resource unit is free.

29. The processor of claim 28 wherein the at least one of the threads has highest priority code among a set of ready threads.

30. The processor of claim 28 wherein the priority selector iteratively assigns resource units to threads in the set of ready threads according to the corresponding priority codes and resource availability until the set becomes empty.

IX. Evidence Appendix

To the best of the knowledge of Appellant, Appellant's legal representative, and Appellant's assignee, there has been no evidence submitted pursuant to 37 C.F.R. §§ 1.130, 1.131, or 1.132, nor has any other evidence been entered in the record by the Examiner and relied upon in this Appeal Brief.

X. Related Proceedings Appendix

To the best of the knowledge of Appellant, Appellant's legal representative, and Appellant's assignee, there are no other appeals or interferences which will directly affect or be directly affected or have a bearing on a decision by the Board of Patent Appeals and Interferences ("the Board") in the pending appeal.

531566_1.DOC